Comparative Analysis of Large Language Model as Feature Extraction Methods In Sarcasm Detection Using Classification Algorithms

1st Darrel Nathaniel Sabera
Information System Study Program
Universitas Multimedia Nusantara
Tangerang, Indonesia
darrel.nathaniel@student.umn.ac.id

2nd* Dinar Ajeng Kristiyanti
Information System Study Program
Universitas Multimedia Nusantara
Tangerang, Indonesia
dinar.kristiyanti@umn.ac.id*

Abstract*— Sarcasm Detection is an important task in Natural Language Processing (NLP) because sarcasm expression can distort sentiment analysis and mislead autamated decision making system. The urgency of this research lies in the limitations of traditional methods such as Bag of Words (BoW) and TF-IDF, which fail to capture deep contextual understanding, while word embedding techniques like Word2Vec and Glove have improvement, difficulties remained in comprehending full sentence meanings. Large Language Model (LLM) such as BERT and RoBERTa have Transformed NLP by capturing contextual word representation, making them more effective for sarcasm detection. This study presents a comparative analysis of various feature extraction method like Word2Vec, Glove, BERT, and RoBERTa combined with classification algorithms such as Support Vector Machine (SVM), XGBoost, and Random Forest. This study uses the Knowledge Discovery in Database (KDD) Framework which includes data selection, preprocessing, transformation, modeling, and evaluation. The dataset consists of news headlines labeled as sarcastic or non sarcastic, Principal Component Analysis used for dimensionality reduction by eliminating redundant features. The result show that the RoBERTa-SVM combination achieves the highest accuracy of 88.00%, indivating the superiority of transformer based models over traditional embedding techniques. This study confludes that integrating contextual embeddings and feature selection improves sarcasm detection perfomance while maintaining computational efficiency. However, the model still faces challenges in identifying implicit sarcasm due the absence of explicit linguistic cuesm which represent a limitation of this research.

Keywords— *Feature Extraction, Feature Selection, Large Language Model, RoBERTa–Support Vector Machine, Sarcasm Detection*

I. INTRODUCTION

In the era of social media and digital communication, sarcasm became a common phenomenon, with the study that showed around 23% of online interaction contained sarcasm expression. Because sarcasm could be an effective tool for criticism, irony, or humor, the misinterpretation often led to misunderstanding, conflicts, and even misinformation. In digital communication, where there was no tone and expression, sarcasm distorted sentiment analysis, affected public opinion, and misled automated decision-making systems in domains such as customer service, social media monitoring, and political discourse [1][2]. This ambiguity made it challenging to differentiate between genuine and sarcastic statements not only for humans but also for Natural Language Processing (NLP) models [3]. Because of these far-reaching implications, accurately detecting sarcasm became an important task in NLP research.

To overcome this challenge, various sarcasm detection methods were explored. Traditional methods like Bag of Words (BoW) and Term Frequency–Inverse Document Frequency (TF-IDF) failed to capture deeper contextual meaning [4]. Word embedding models like Word2Vec and GloVe improved upon traditional approaches, but processing entire sentence meanings required for sarcasm detection remained a challenge [5]. The emergence of Large Language Models (LLMs) such as BERT, GPT, and T5 transformed NLP tasks by incorporating deep contextual embeddings, allowing them to better distinguish between genuine and sarcastic meanings in sarcasm detection [6]. Studies have demonstrated that BERT based model outperform traditional method, achieving up to 85% accuracy in sarcasm detection task [7]. The transformer-based model used an attention mechanism to analyze the surrounding context to improve sarcasm classification accuracy [8]. However, despite its many advantages, LLMs faced challenges such as high computational cost and interpretability issues, making real-world implementation complex [9].

Since sarcasm often relied on elements like hyperbole, irony, and contradiction, sarcasm detection required advanced feature extraction methods that went beyond simple word matching [10]. LLM-based feature extraction techniques demonstrated superior performance due to their ability to process large amounts of data and understand complex interactions between words [11]. Although these models enhanced detection accuracy, significant computational resources were required, leading to a trade-off between performance and efficiency. To classify sarcasm more effectively, machine learning models such as XGBoost, Support Vector Machine (SVM), and Random Forest were often used. These classifiers leveraged extracted features to distinguish between sarcastic and non-sarcastic statements [12].

The ability to detect sarcasm had important applications in sentiment analysis, product reviews, and social media monitoring. Poor handling of sarcasm can introduce bias and lead to misleading conclusions, especially in automated decision-making systems [13]. To address these challenges, this research proposed a comparative study of feature extraction methods in sarcasm detection, focusing on traditional techniques such as Word2Vec and GloVe versus modern transformer-based models such as BERT and RoBERTa. By analyzing their performance when combined with Support Vector Machine (SVM), XGBoost, and Random Forest, this study aimed to determine the most effective method for sarcasm classification while balancing accuracy and computational efficiency. Implementing dimensionality reduction techniques such as Principal Component Analysis (PCA) helped mitigate computational demands while maintaining model effectiveness. By integrating advanced NLP models with optimized feature selection, this study aim to contribute to the development of more robust and adaptive sarcasm detection.

II. RELATED WORKS

Sarcasm detection research advanced due to advancements in feature extraction methods and machine learning models. Traditional methods, such as TF-IDF and BoW, struggled with assessing semantic meaning in sarcastic language samples. Word2Vec and GloVe addressed these issues by addressing semantic deficiencies in word representation models, resulting in an average accuracy of 70% or less on social media data sources [14].

Word2Vec and GloVe were superior to conventional methods but had a 75% success rate for sarcasm detection. Large Language Models like BERT, RoBERTa, and T5 made significant progress in NLP. Transformer-based approaches viewed text from both sides, resulting in rich contextual feature representations. BERT achieved 85% accuracy in sarcasm recognition on social media datasets[15]. With an accuracy of 76% through intensive training on a large amount of data, RoBERTa showed higher performance as an improvement over BERT [7]. LLM-based models demonstrated superior ability to assess both actual and implied meanings in sarcastic texts, surpassing conventional embedding techniques including Word2Vec and GloVe [4].

This research focused on the field of study regarding the combination of feature extraction techniques with classification models. After feature extraction, machine learning models including Support Vector Machine (SVM) and XGBoost combined with Random Forest operated for classification purposes [16]. The Word2Vec and SVM methods effectively identified sarcasm in Twitter content, with an accuracy of 72% [17]. Two LLM-based approaches, BERT or RoBERTa, used alone with SVM showed a higher accuracy result around 75%, indicating their better capacity to identify patterns in complex text [18].

The analysis of traditional methods compared to LLM-based methods requires a comparative study for performance evaluation. LLMs provided better results than traditional approaches while showing an improvement in accuracy levels of 10–15% over standard protocols for the ability to generalize new data [19]. The enhanced performance from LLM-based methods typically required additional computational resources as well as significant resource needs. Research should have explored how accuracy related to computational cost because these parameters needed to be evaluated.

Detection of sarcasm impacted many practical fields, including product sentiment evaluation and public sentiment tracking on social media. A study showed how feature extraction using the LLM method improved the accuracy of sentiment analysis in social media evaluation along with product review analysis [20].

The main objective of this research was to evaluate several feature extraction methods for sarcasm detection. This research compared Word2Vec and GloVe with BERT and RoBERTa to identify the strengths and weaknesses of each before recommending the best solution for the classification model..

III. RESEARCH METHOD

The study employs the Knowledge Discovery Database (KDD) framework for detecting sarcasm, involving data selection, preprocessing, feature extraction, and modeling [21]. Preprocessing removes duplicates and handles missing values. Word embeddings are used for feature extraction. Modeling uses SVM, XGBoost, and Random Forest to identify sarcastic text. Performance metrics evaluate the model's effectiveness [22]. Figure 1 show the workflow for this research.

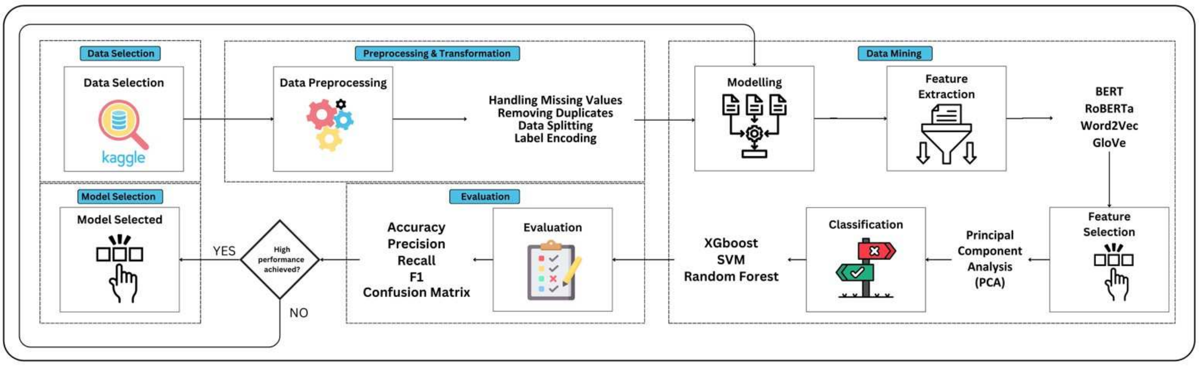

Fig. 1. Research Workflow

3.1 Data Selection

Data were collected from Kaggle website for Sarcastic news headlines. This dataset was obtained from The Onion and HuffPost (https://www.kaggle.com/datasets/rmisra/news-headlinesdataset-for-sarcasm-detection/data). It comprised both sarcastic and regular news headlines. The sarcastic headlines are obtained from The Onion and regular headlines are obtained from HuffPost [23]. Due to its writing is not from general population, the chance of spelling errors and informal language was very low. The dataset contain 27,000 headlines, as of it 11,700 are sarcastic and 14,900 are non sarcastic. From. The dataset consisted of 27,000 headlines, with 11,700 sarcastic and 14,900 non-sarcastic. 10,000 were used, with 8,000 for training and 2,000 for testing. Table 1 displays sample statements from The Onion and HuffPost.

TABLE I. EXAMPLE OF SARCASM DATA

Text	Sarcastic ID
nasa now almost positive mars is rocky	True
longtime teacher retires without changing a single student's life	True

donald trump heading for a series of wins in the northeast, polls say	False
new 'star-wars' film once again disappoints die-hard nien nunb fans	True
ryan lochte apologizes for behavior in rio	False
4 lessons prison taught me about power and control	False

3.2 Data Preprocessing and Transformation

Data preprocessing is an important step in data analysis, involving handling missing values, removing duplicates, data splitting, and label encoding. These steps prevent bias, maintain data integrity, eliminate redundancy, ensure proper training and evaluation, and improve model accuracy and reliability [24].

1. *Missing Values*

Missing values in data were crucial for machine learning models' quality and reliability. Caused by collection failures, user deletions, or system faults, such missing entries could have distorted statistical analyses, reduced model accuracy, and introduced biases. Solutions like imputation or removal were used to mitigate these issues [24].

2. *Removing Duplicates*

Duplicates in model development could have led to erroneous evaluations, deceptive performance indicators, and biases. Inadequate management often resulted in overfitting and skewed feature distributions. Eliminating duplicates, enhanced model accuracy, consistency, and robustness, resulting in more reliable predictions and improved decision-making [23].

3. *Data Splitting*

Data splitting is a technique used to divide a dataset into subsets for training and evaluation, with a specific portion allocated for model training and the remaining for testing. For example, in a sarcasm detection study, an 80:20 ratio was used to represent both sarcastic and non-sarcastic classes through stratified sampling [23].

4. *Label Encoding*

Label encoding is a crucial technique in machine learning, as it converts categorical values into numerical representations, assigning a unique label to each variable, especially for mathematical algorithms [24]. By using label encoding, categories such as “sarcastic” and “non-sarcastic” were converted into numerical values, for example, 1 for sarcasm and 0 for non-sarcasm.

3.3 Feature Extraction using Large Language Model

Feature extraction is crucial for processing unstructured data like text, images, audio, and video. It converted the data into vectors, known as embeddings, which could not be directly processed using algorithms. This research compared transformer models like BERT and RoBERTa with standard embeddings like GloVe and Word2Vec [11].

1. *Transformers Model*

Transformers is deep learning architectures that enhanced natural language processing by using self-attention methods to parse full sequences. Local and global dependencies in text were captured through attention scores that identified word correlations and multi-head attention to captured contextual information. Positional encoding is used to integrate word order, reducing repetition in Transformers [25].

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (1)$$

Equation (1) defines the attention mechanism, whereby Q, K, and V signify the query, key, and value matrices, respectively. Multi-head attention improves this by calculating several independent attention heads, each representing distinct aspect of the information. The results are subsequently combined, yielding a more enriched and comprehensive representation.

Transformers like BERT and RoBERTa can identify sarcasm by understanding both explicit and implicit language meanings, learning complex verbal nuances, and differentiate sarcastic from literal utterances. Contextual word representations are constructed by considering both sides of a word [8].

2. *Word2Vec*

Word2Vec is a deep learning model that converts words into dense vector representations using the Continuous Bag-of-Words (CBOW) or Skip-gram architecture. It is used in sarcasm detection research to understand semantic relationships between headline words. [26].

3. *GloVe*

GloVe is a matrix-based approach that used global statistics to create word vector representations. It optimized representation by considering the frequency of co-occurring words, unlike Word2Vec, which focused on local context. This study used GloVe to better understand the contextual meaning of words, which is crucial for identifying sarcasm patterns [27].

3.4 Feature Selection

Principal Component Analysis (PCA) is a statistical technique used in feature selection to reduce dimensionality by transforming high-dimensional data into principal components, preserving the greatest variance. It enhanced model efficacy and interpretability by addressing multicollinearity and reducing overfitting, making it widely used in machine learning, pattern identification, and image processing [28].

3.5 Classification

This research presented and implemented three effective text categorization algorithms: SVM, XGBoost, and Random Forest, each proven to accurately handle, analyze, and categorize textual data in machine learning and natural language processing. Experiments and performance assessments demonstrated these models could handle various text classification challenges [7].

1. *Support Vector Machine (SVM)*

SVM is a machine learning algorithm used for classification tasks by finding the optimal hyperplane that separated data into different classes [29]. The SVM algorithm, when combined with deep feature extraction techniques, effectively handles complex data and distinguishes between sarcastic and non-sarcastic headlines, ensuring accurate decisions in high-dimensional feature spaces. [30].

$$\hat{y}_i = sign\left(\sum_{i=1}^{N} \alpha_i\, y_i K(x_i, x) + b\right) \quad (2)$$

The Equation (2) showed the decision boundaries function for SVM with kernel function. The predicted class $\hat{y}$ of a new data point x by taking the sign of a weighted sum of kernel evaluation between x and each support vector x_i. α_i are the learned coefficients, y_i are the corresponding class labels (+1 or -1), and $K(x_i, x)$ is the kernel function to measure similarity between x and x_i [31].

2. XGboost

XGBoost is a boosting algorithm that built models incrementally by optimizing the loss function and adding simple models to correct previous errors. It is known for its efficiency, speed, and high accuracy, particularly on complex datasets with intricate feature interaction patterns. In sarcasm detection, it helped capture subtle patterns and enhanced classification performance [32].

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i) \quad (3)$$

The equation (3) showed how XGBoost makes predictions by summing the outputs of K individual decision trees. Each tree f_k takes the input x_i and produces a prediction, the final output $\hat{y}_i$ is the total of these individual predictions. By fixing mistakes from prior trees, this additive method lets XGBoost progressively increase accuracy [33].

3. Random Forest

Random Forest is an ensemble method that created multiple decision trees using various data and features, with each tree offering a forecast based on majority voting. This method reduced the risk of overfitting errors and strengthened [34]. It is used in sarcasm detection studies to manage feature diversity, making the model more flexible in changing sarcastic language usage.

$$F(x) = \frac{1}{T}\sum_{t=1}^{T} h_t(x) \quad (4)$$

Equation (4) showed the decision boundaries function of a Random Forest, where $F(x)$ is the average prediction from all T decision trees for input x. Each $h_t(x)$ denotes the output of the t-th tree, typically 0 or 1 for binary classification. The final class is determined by majority vote [35].

3.6 Evaluation

The evaluation of a sarcasm detection model is based on essential metrics obtained from the confusion matrix: accuracy, precision, recall, and F1 score. These metrics offered a thorough evaluation of the model's capability to differentiate sarcasm from non-sarcasm [36]. Equation (5) delineated accuracy. Precision measured the ratio of accurately identified sarcastic instances to the total cases anticipated as sarcastic. A high precision score signified that the model reduced false positives. (6) delineated precision. Recall (or sensitivity) measured the model's capacity to detect all genuine occurrences of sarcasm. It quantified the model's efficacy in identifying affirmative cases, as illustrated in (7). The F1 score offered a fair assessment of precision and recall, rendering it especially advantageous for imbalanced datasets. The harmonic mean of precision and recall is articulated in (8).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \quad (8)$$

IV. RESULT AND DISCUSSION

4.1 Comparison of Large Language Model Results without Feature Selection

Table II specified the effectiveness of several models for sarcasm detection in combination with classifiers such as XGBoost (XGB), Support Vector Machine (SVM), and Random Forest (RF). Of all the combinations, RoBERTa-SVM had the greatest accuracy at 86.00%, followed by BERT-SVM and BERT-XGB, both at 84.00%, suggesting that SVM works best with transformer-based embeddings. While GloVe-based models, employing just 25 features, fared the worst, with Glove-SVM only reaching 69.00% accuracy, Word2Vec-based models exhibited reasonable performance with accuracies around 76.00%. Models using transformer-based embeddings (RoBERTa and BERT) clearly outperformed those using conventional word embeddings overall, hence stressing their ability to capture contextual subtleties for sarcasm identification.

TABLE II. COMPARISON OF LARGE LANGUAGE MODEL RESULTS WITHOUT FEATURE SELECTION

Model	**No. Features**	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**
BERT-XGB	768	84.00	85.00	83.00	83.00
BERT-SVM		84.00	84.00	83.00	84.00
BERT-RF		82.00	82.00	80.00	81.00
RoBERTa-XGB	768	83.00	82.00	82.00	82.00
RoBERTa-SVM		86.00	85.00	86.00	85.00
RoBERTa-RF		79.00	80.00	77.00	78.00
Word2Vec-XGB	300	76.00	76.00	74.00	75.00
Word2Vec-SVM		76.00	76.00	74.00	75.00
Word2Vec-RF		73.00	74.00	68.00	71.00
Glove-XGB	25	74.00	73.00	72.00	73.00
Glove-SVM		69.00	70.00	65.00	67.00
Glove-RF		72.00	73.00	66.00	70.00

4.2 Comparison of Large Language Model Results using Feature Selection

The study gave insights into the effects of using PCA on the performance of the models when the number of features is reduced. It turned out that PCA made a few sets of models better and caused only small decreases in others. Using PCA based feature selection as shown in Table III leads to a higher model performance by eliminating any redundant aspects yet preserving the main features. According to the results, RoBERTa-SVM surpassed BERT-SVM with an accuracy of 88.00% as opposed to 86.00%, The study reveals that transformer embeddings with reduced dimensionality yield significant improvements, with models like Word2Vec and GloVe showing gains but lower overall scores, indicating PCA can perform faster calculations and make more accurate predictions..

TABLE III. COMPARISON OF LARGE LANGUAGE MODEL RESULTS USING FEATURE SELECTION

Model	No. Features	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
BERT-XGB	624	81.00	81.00	80.00	80.0
BERT-SVM		86.00	85.00	87.00	86.00
BERT-RF		80.00	82.00	75.00	78.00
RoBERTa-XGB	433	80.00	80.00	79.00	79.00
RoBERTa-SVM		88.00	87.00	88.00	87.00
RoBERTa-RF		79.00	80.00	75.00	78.00
Word2Vec-XGB	284	77.00	79.00	72.00	76.00
Word2Vec-SVM		79.00	80.00	78.00	79.00
Word2Vec-RF		75.00	77.00	68.00	72.00
Glove-XGB	23	73.000	74.00	69.00	71.00
Glove-SVM		66.00	66.00	65.00	65.00
Glove-RF		72.00	73.00	66.00	69.00

From the matrix shown in Figure 2, the model correctly classified 903 True Positive samples and 859 True Negative samples, but made mistakes in 124 False Positive samples and 114 False Negative samples. Calculating using the confusion matrix Accuracy formula, the model achieved 88.1% accuracy, despite some mistakes.

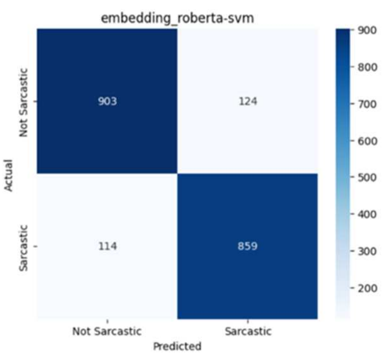

Fig. 2. Confusion Matrix RoBERTa-SVM Model Based using PCA

4.3 Discussion

Transformer-based architectures like BERT and RoBERTa outperformed traditional models like Word2Vec and GloVe in detecting sarcasm. However, challenges remained in identifying subtle forms of sarcasm, particularly those relying on implicit cues. The study suggests Principal Component Analysis (PCA) could enhance model efficiency by reducing feature space dimensionality, removing redundant features, improving generalization, and increasing training speed, scalability, and noise. PCA also improved performance by 2% through optimized features and improved computational efficiency.

TABLE IV. ACCURACY COMPARISON RESULTS FROM PREVIOUS RESEARCH

Research	Accuracy (%)		
	Proposed Model	[7]	[18]
BERT-SVM Model (Without Feature Selection)	84.00	67.60	74.50
RoBERTa–SVM (Without Feature Selection)	86.00	-	76.80
RoBERTa-SVM (Using Feature Selection)	88.10	-	-

Table IV showed a comparison of detection performance between transformer-based models and previous studies, revealing that the RoBERTa-SVM model using feature selection (PCA) achieved the highest accuracy at 88.10%. This improvement was attributed to PCA enhancement of feature representation and reduction of dimensionality, thereby enhancing the overall performance of sarcasm detection models, thereby supporting the effectiveness of integrating advanced transformer-based language models with dimensionality reduction methods [28].

V. CONCLUSION

The study reveals that transformer based model like BERT and RoBERTa outperform traditional word embeddings such as Word2Vec and Glove in sarcasm detection due to their deeper contextual understanding. The RoBERTa-SVM combination achieved the highest accuracy of 88.10% with Principal Component Analysis (PCA), which improved performance by 2.00% through dimensionality reduction without lowering accuracy. The detection of sarcasm faces ongoing difficulties because it requires detection within context and its language usage remains discreet. Future research needs to develop combined models using tone and text to boost detection success rates.

ACKNOWLEDGMENT

The authors would like to thank to Universitas Multimedia Nusantara for the support and resources provided throughout the development of this research.

REFERENCES

[1] A. Ashwitha, G. Shruthi, H. R. Shruthi, M. Upadhyaya, A. P. Ray, and T. C. Manjunath, “Sarcasm detection in natural language processing,” *Mater. Today Proc.*, vol. 37, no. Part 2, pp. 3324–3331, 2020, doi: 10.1016/j.matpr.2020.09.124.

[2] D. Olaniyan, R. O. Ogundokun, O. P. Bernard, J. Olaniyan, R. Maskeliūnas, and H. B. Akande, “Utilizing an Attention-Based LSTM Model for Detecting Sarcasm and Irony in Social Media,” *Computers*, vol. 12, no. 11, pp. 1–15, 2023, doi:

10.3390/computers12110231.

[3] A. Onan and M. A. Tocoglu, "A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification," *IEEE Access*, vol. 9, pp. 7701–7722, 2021, doi: 10.1109/ACCESS.2021.3049734.

[4] C. R. Prasad, N. A. Reddy, G. R. Varma, and M. Shuaib, "Sarcasm Detection With Glove and Word2Vec Models," *ARPN J. Eng. Appl. Sci.*, vol. 18, no. 10, pp. 1181–1186, 2023, doi: 10.59018/0523154.

[5] N. Babanejad, H. Davoudi, A. An, and M. Papagelis, "Affective and Contextual Embedding for Sarcasm Detection," *COLING 2020 - 28th Int. Conf. Comput. Linguist. Proc. Conf.*, pp. 225–243, 2020, doi: 10.18653/v1/2020.coling-main.20.

[6] D. Kumar and S. Singh, "Advancements in Transformer Architectures for Large Language Model: From Bert To Gpt-3 and Beyond," *Int. Res. J. Mod. Eng. Technol. Sci.*, no. May, 2024, doi: 10.56726/irjmets55985.

[7] D. Sandor and M. Bagic Babac, "Sarcasm detection in online comments using machine learning," *Inf. Discov. Deliv.*, vol. 52, no. 2, pp. 213–226, 2024, doi: 10.1108/IDD-01-2023-0002.

[8] J. Dai, "A BERT-Based with Fuzzy logic Sentimental Classifier for Sarcasm Detection," *2024 7th Int. Conf. Adv. Algorithms Control Eng. ICAACE 2024*, vol. 0, pp. 1275–1280, 2024, doi: 10.1109/ICAACE61206.2024.10548550.

[9] A. Kumar and V. Anand, "Transformers on sarcasm detection with context," *Proc. Annu. Meet. Assoc. Comput. Linguist.*, pp. 88–92, 2020, doi: 10.18653/v1/P17.

[10] Z. L. Chia, M. Ptaszynski, F. Masui, G. Leliwa, and M. Wroczynski, "Machine Learning and feature engineering-based study into sarcasm and irony classification with application to cyberbullying detection," *Inf. Process. Manag.*, vol. 58, no. 4, 2021, doi: 10.1016/j.ipm.2021.102600.

[11] A. Kumar, V. T. Narapareddy, P. Gupta, V. A. Srikanth, L. B. M. Neti, and A. Malapati, "Adversarial and Auxiliary Features-Aware BERT for Sarcasm Detection," *ACM Int. Conf. Proceeding Ser.*, pp. 163–170, 2020, doi: 10.1145/3430984.3431024.

[12] V. Prayag, "Optimal Feature Extraction based Machine Learning Approach for Sarcasm Type Detection in News Headlines," *Int. J. Comput. Appl.*, vol. 177, no. 46, pp. 25–29, 2020, doi: 10.5120/ijca2020919981.

[13] M. V. Rao and C. Sindhu, "Detection of Sarcasm on Amazon Product Reviews using Machine Learning Algorithms under Sentiment Analysis," *2021 Int. Conf. Wirel. Commun. Signal Process. Networking, WiSPNET 2021*, pp. 196–199, 2021, doi: 10.1109/WiSPNET51692.2021.9419432.

[14] A. Khatri and P. Pranav, "Sarcasm detection in tweets with BERT and GloVe embeddings," *Proc. Annu. Meet. Assoc. Comput. Linguist.*, pp. 56–60, 2020, doi: 10.18653/v1/P17.

[15] M. A. Bagioyuwono, D. A. Kristiyanti, A. Sony, and E. Nugroho, "Designing an E-Repository of Sentiment Data and Cyberbullying Detection in Indonesian using a Parameter Optimization Algorithm for LSTM," pp. 0–1, 2024.

[16] D. A. Kristiyanti and Sri Hardani, "Sentiment Analysis of Public Acceptance of Covid-19 Vaccines Types in Indonesia using Naïve Bayes, Support Vector Machine, and Long Short-Term Memory (LSTM)," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 7, no. 3, pp. 722–732, 2023, doi: 10.29207/resti.v7i3.4737.

[17] R. Gupta, J. Kumar, H. Agrawal, and Kunal, "A Statistical Approach for Sarcasm Detection Using Twitter Data," *Proc. Int. Conf. Intell. Comput. Control Syst. ICICCS 2020*, no. Iciccs, pp. 633–638, 2020, doi: 10.1109/ICICCS48265.2020.9120917.

[18] X. Shu and S. Diego, "BERT and RoBERTa for Sarcasm Detection : Optimizing Performance through Advanced Fine-tuning," vol. 0, pp. 1–11, 2024, doi: 10.54254/2755-2721/97/20241354.

[19] J. Yuan, R. Tang, X. Jiang, and X. Hu, "Large Language Models for Healthcare Data Augmentation: An Example on Patient-Trial Matching," *AMIA ... Annu. Symp. proceedings. AMIA Symp.*, vol. 2023, pp. 1324–1333, 2023.

[20] D. K. Sharma, B. Singh, S. Agarwal, N. Pachauri, A. A. Alhussan, and H. A. Abdallah, "Sarcasm Detection over Social Media Platforms Using Hybrid Ensemble Model with Fuzzy Logic," *Electron.*, vol. 12, no. 4, pp. 1–21, 2023, doi: 10.3390/electronics12040937.

[21] S. Khlamov, V. Savanevych, O. Briukhovetskyi, I. Tabakova, and T. Trunova, "Data Mining of the Astronomical Images by the CoLiTec Software," *CEUR Workshop Proc.*, vol. 3171, pp. 1043–1055, 2022.

[22] T. S. Et. al., "Sarcasm Detection From Twitter Database Using Text Mining Algorithms," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 11, pp. 1916–1924, 2021, doi: 10.17762/turcomat.v12i11.6144.

[23] R. Misra and P. Arora, "Sarcasm detection using news headlines dataset," *AI Open*, vol. 4, no. October 2022, pp. 13–18, 2023, doi: 10.1016/j.aiopen.2023.01.001.

[24] I. Chahid, A. K. Elmiad, and M. Badaoui, "Data Preprocessing For Machine Learning Applications in Healthcare: A Review," *Proc. - SITA 2023 2023 14th Int. Conf. Intell. Syst. Theor. Appl.*, pp. 1–6, 2023, doi: 10.1109/SITA60746.2023.10373591.

[25] R. K. Singh, "Advancements in Natural language Processing: An In-depth Review of Language Transformer Models," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 12, no. 6, pp. 1719–1732, 2024, doi: 10.22214/ijraset.2024.63408.

[26] A. Annie Johnson and R. Karthik, "Performance Evaluation of Word Embeddings for Sarcasm Detection- A Deep Learning Approach," 2021, doi: 10.4108/eai.16-5-2020.2303935.

[27] C. I. Eke, A. Norman, L. Shuib, F. B. Fatokun, and I. Omame, "The Significance of Global Vectors Representation in Sarcasm Analysis," *2020 Int. Conf. Math. Comput. Eng. Comput. Sci. ICMCECS 2020*, 2020, doi: 10.1109/ICMCECS47690.2020.246997.

[28] B. M. S. Hasan and A. M. Abdulazeez, "A Review of Principal Component Analysis Algorithm for Dimensionality Reduction," *J. Soft Comput. Data Min.*, vol. 2, no. 1, pp. 20–30, 2021, doi: 10.30880/jscdm.2021.02.01.003.

[29] L. R. Halim and A. Suryadibrata, "Cyberbullying Sentiment Analysis with Word2Vec and One-Against-All Support Vector Machine," *IJNMT (International J. New Media Technol.*, vol. 8, no. 1, pp. 57–64, 2021, doi: 10.31937/ijnmt.v8i1.2047.

[30] S. M. Sarsam, H. Al-Samarraie, A. I. Alzahrani, and B. Wright, "Sarcasm detection using machine learning algorithms in Twitter: A systematic review," *Int. J. Mark. Res.*, vol. 62, no. 5, pp. 578–598, 2020, doi: 10.1177/1470785320921779.

[31] K. L. Du, B. Jiang, J. Lu, J. Hua, and M. N. S. Swamy, "Exploring Kernel Machines and Support Vector Machines: Principles, Techniques, and Future Directions," *Mathematics*, vol. 12, no. 24, pp. 1–58, 2024, doi: 10.3390/math12243935.

[32] A. Kumar, S. Dikshit, and V. H. C. Albuquerque, "Explainable Artificial Intelligence for Sarcasm Detection in Dialogues," *Wirel. Commun. Mob. Comput.*, vol. 2021, 2021, doi: 10.1155/2021/2939334.

[33] M. Imani, A. Beikmohammadi, and H. R. Arabnia, "Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels," *Technologies*, vol. 13, no. 3, pp. 1–40, 2025, doi: 10.3390/technologies13030088.

[34] C. Eke, A. A. Norman, L. Shuib, F. Faith B., and Z. A. Long, "Random Forest-Based Classifier for Automatic Sarcasm Classification on Twitter Data Using Multiple Features," *J. Inf. Syst. Digit. Technol.*, vol. 4, no. 2, pp. 125–145, 2022, doi: 10.31436/jisdt.v4i2.345.

[35] H. Dabiri, V. Farhangi, M. J. Moradi, M. Zadehmohamad, and M. Karakouzian, "Applications of Decision Tree and Random Forest as Tree-Based Machine Learning Techniques for Analyzing the Ultimate Strain of Spliced and Non-Spliced Reinforcement Bars," *Appl. Sci.*, vol. 12, no. 10, pp. 1–13, 2022, doi: 10.3390/app12104851.

[36] A. Y. Abdullah Amer and T. Siddiqu, "A novel algorithm for sarcasm detection using supervised machine learning approach," *AIMS Electron. Electr. Eng.*, vol. 6, no. 4, pp. 345–369, 2022, doi: 10.3934/electreng.2022021.